Optimizing Experiments with Pyomo.DoE

`dowlinglab.github.io/pyomo-doe`

Alexander (Alex) Dowling, Ph.D.
Daniel Laky, Ph.D.
Chemical and Biomolecular Engineering
University of Notre Dame

Advanced PSE+ Stakeholder Summit
September 19, 2024
Pittsburgh, PA

PYOMO DoE

Contributors to Pyomo.DoE and ParmEst: Jialu Wang (ND), Dan Laky (ND), Hailey Lynch (ND), John Siirola (SNL), Bethany Nicholson (SNL), Miranda Mundt (SNL), Shawn Martin (SNL), Katherine Klise (SNL)

UNIVERSITY OF NOTRE DAME

CCSI[2] Carbon Capture Simulation for Industry Impact

PROMMIS Process Optimization and Modeling for Minerals Sustainability

Ask us for a Pyomo.DoE pin!

Power of Adaptive Sequential Optimal Experiments

Self-Driving Laboratories

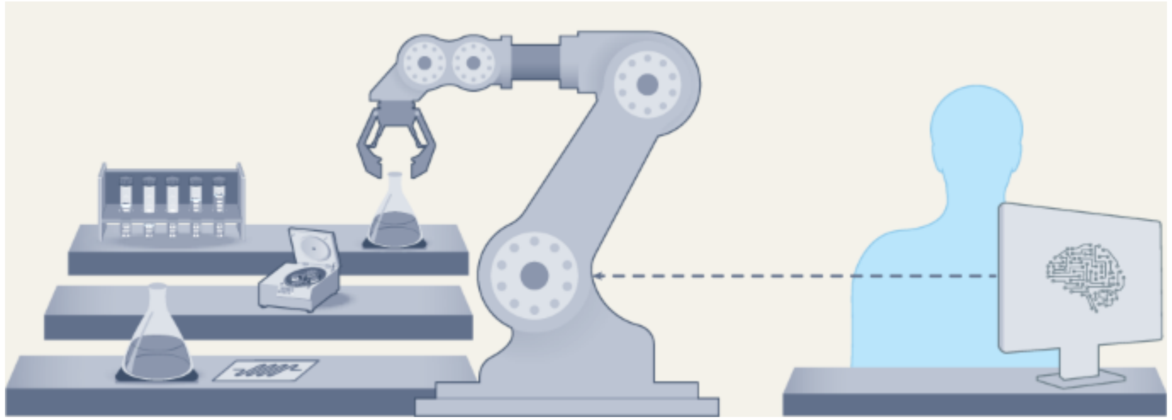
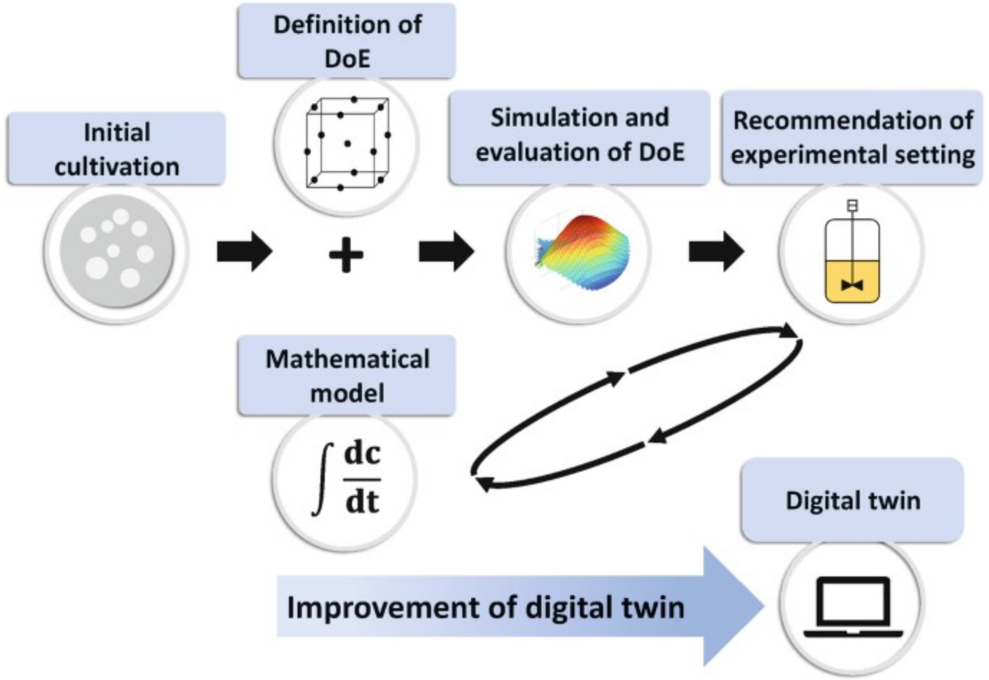


Figure: Abolhasani & Kumacheva (2023), *Nature Syn.*

Epps et al. (2022), *Advanced Materials*
MacLeod et al. (2020), *Science Advances*
MacLeod et al. (2022), *Nature Communications*
Hase, Roch, Aspuru-Guzik (2019), *Trends in Chemistry*
Seifrid et al. (2022), *Acc. Chem. Res.*

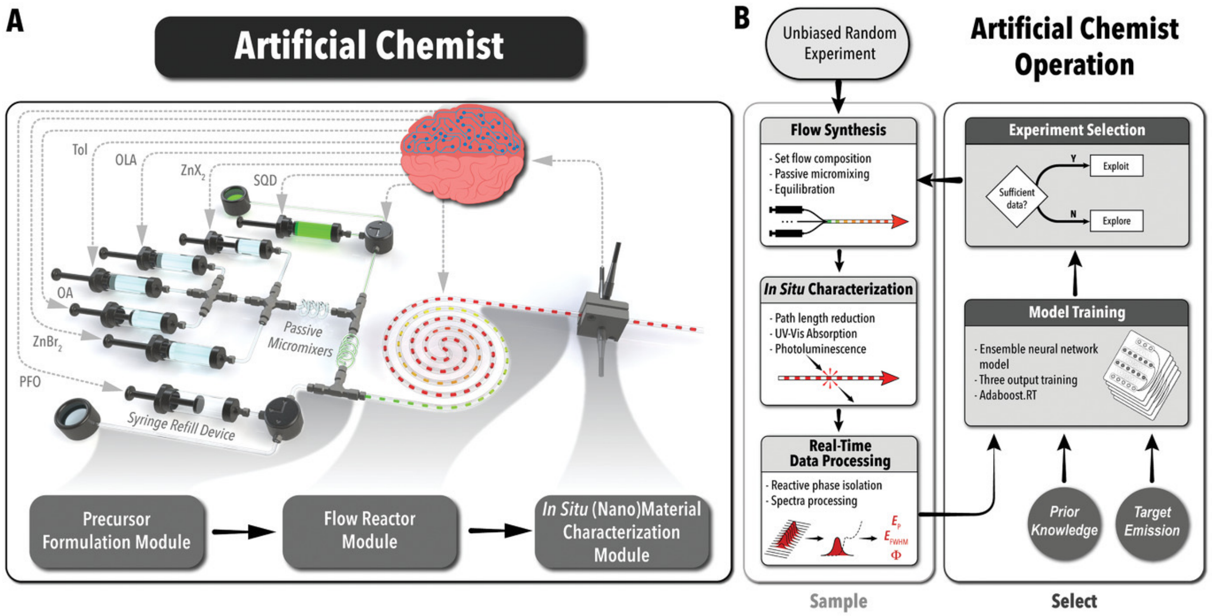
Automation + Model-Based Design of Experiments



Kuchemuller et al. (2020), *Digital Twins*

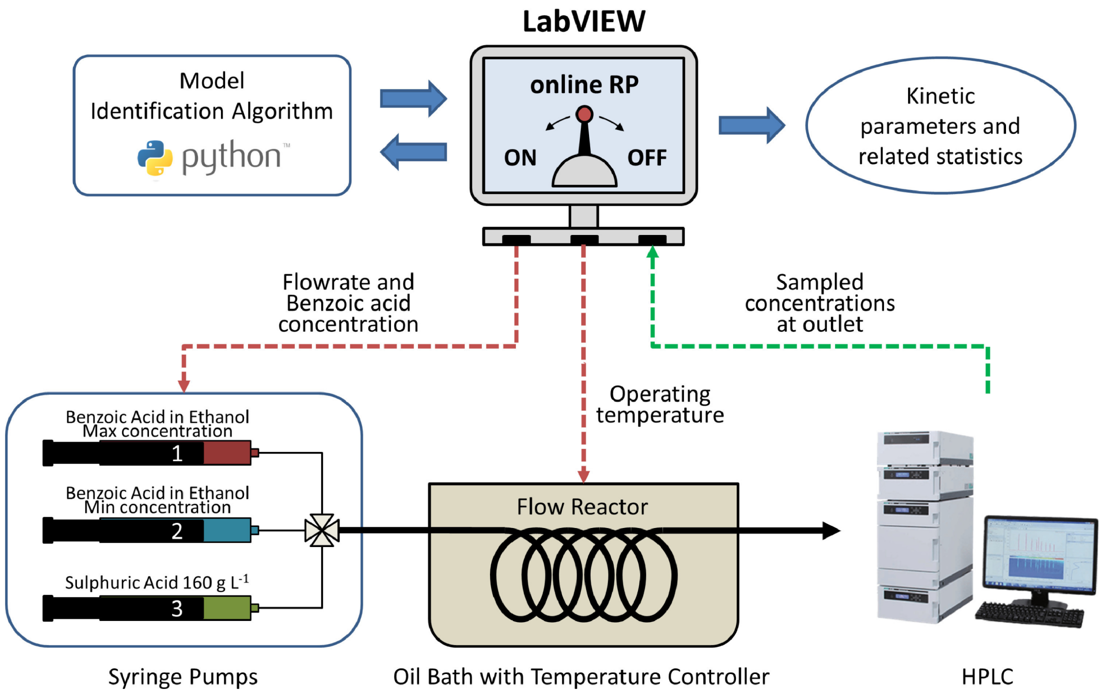
Many Recent Examples of Sequential Optimal Experiments

Quantum Dots
(Machine Learning)



Epps et al. (2022), *Advanced Materials*

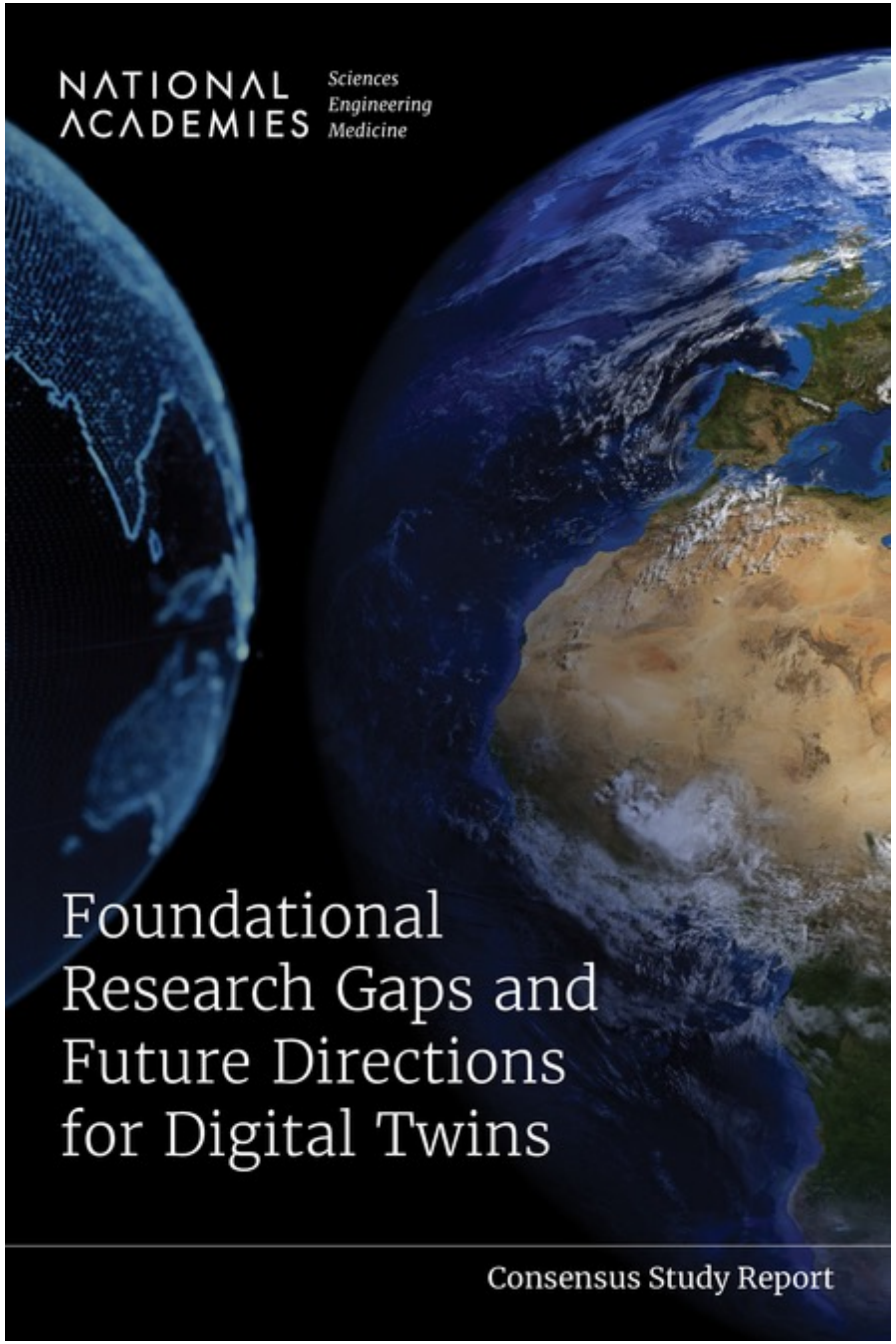
Reaction Engineering
(Science-based Models)



Quaglio et al. (2019), *Comp. & Chem. Eng.*

What is a Digital Twin?

*A **digital twin** is a set of **virtual information constructs** that mimics the structure, context, and behavior of a **natural**, **engineered**, or **social system** (or **system-of-systems**), is **dynamically updated** with data from its physical twin, has a **predictive capability**, and **informs decisions that realize value**. The **bidirectional interaction** between the virtual and the physical is central to the digital twin.*



Science-based Data Analytics Workflow

Prior Knowledge

Data

Model(s)

Parameter Estimation

Model Discrimination

Sensitivity & Uncertainty Analysis

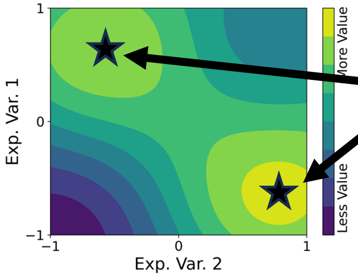
Do I have a good model?

No

Optimize New Experiment(s)

Yes

Model with quantified uncertainty



Optimal new Experiment(s)

Goal: iteratively develop and validate predictive science-based models (**digital twins**)

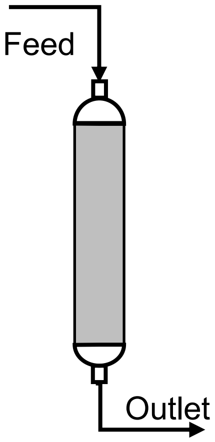
UNIVERSITY OF NOTRE DAME

CCSI²
Carbon Capture Simulation for Industry Impact

PROMMIS
Process Optimization and Modeling for Minerals Sustainability

SBDoE (a.k.a. MBDoE) Facilitates Collaborations

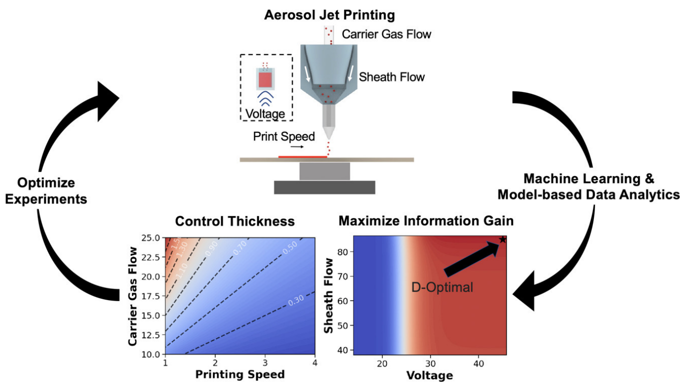
CO_2 Capture



Jialu Wang

Wang, J. and Dowling, A.W. (2022), *AIChE J.* e17813.

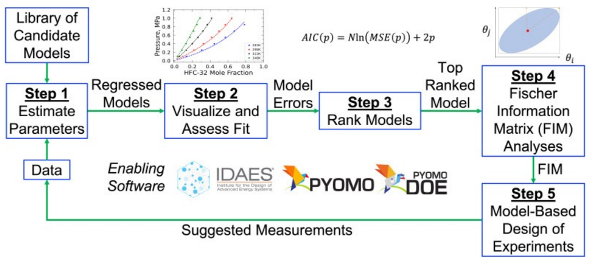
Additive Manufacturing of Thermoelectric Devices



Ke Wang

Wang K., Zhang M., Wang, J., Shang, W., Zhang, Y., Luo, T., Dowling, A.W. (2023), *Digital Chemical Engineering*

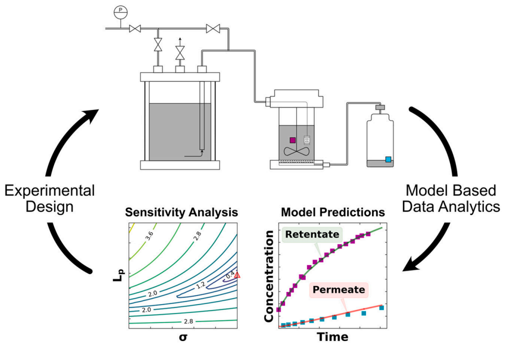
Thermodynamic Modeling (Refrigerants)



Dr. Bridgette Befort

Befort, B.J., Garciadiego, A., Wang, J., Wang, K., Maginn, E.J., Dowling, A.W. (2023), *Fluid Phase Equilibria.*

Rapid/Automated Membrane Characterization



Xinhong Liu

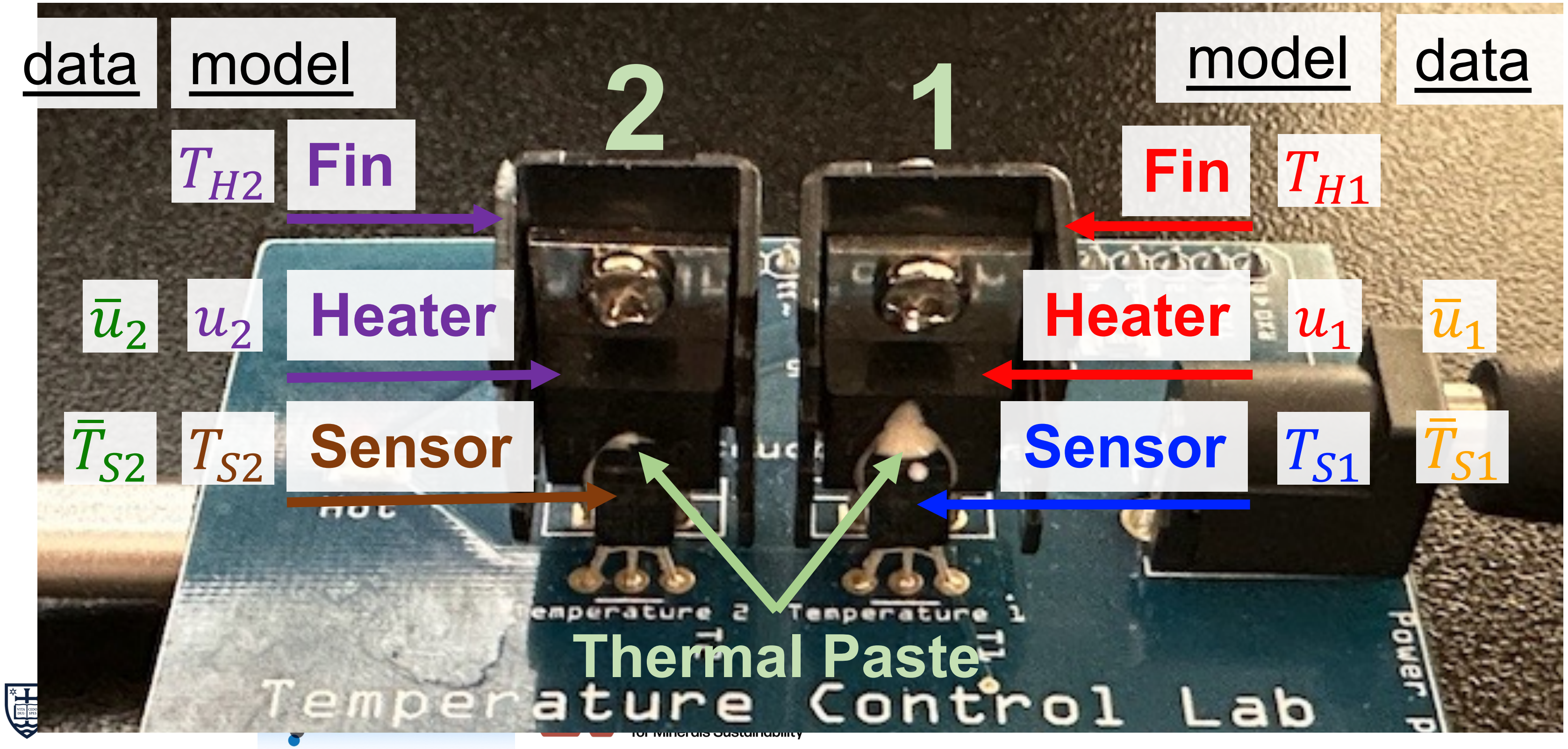
Ouimet, J.A, Xinhong, L., Brown, D.J., Eugene, E.A., Popps, T., Muetzel, Z.W., Dowling, A.W., Phillip, W.A., (2022). *J. Membrane Science.*

Pyomo.DoE Example: Temperature Control Lab (TC Lab)

`dowlinglab.github.io/pyomo-doe/notebooks/tclab_model.html`

$$C_p^H \frac{dT_{H,1}}{dt} = U_a(T_{amb} - T_{H,1}) + U_b(T_{S,1} - T_{H,1}) + \alpha P_1 u_1$$

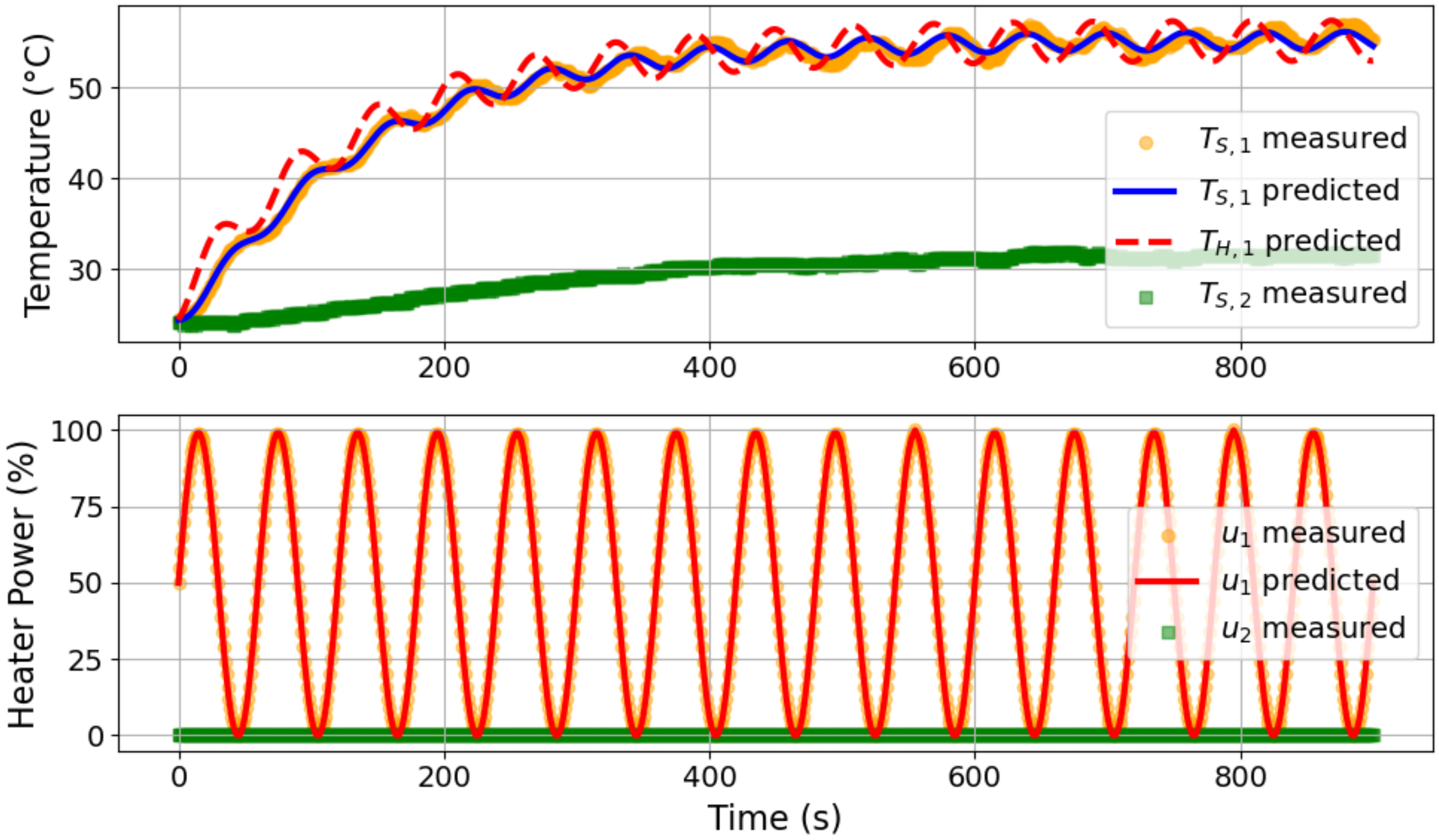
$$C_p^S \frac{dT_{S,1}}{dt} = U_b(T_{H,1} - T_{S,1}), \qquad \theta = (U_a, U_b, C_p^H, C_p^S)^\intercal$$



Thank you to Prof. Jeff Kantor (1954-2023) for the TCLab example and so much more.

TC Lab: Data and Parameter Estimation

Hands-On Tutorial: `dowlinglab.github.io/pyomo-doe`



UNIVERSITY OF NOTRE DAME | CCSI² Carbon Capture Simulation for Industry Impact | PROMMIS Process Optimization and Modeling for Minerals Sustainability

Notebooks at `dowlinglab.github.io/pyomo-doe`

TCLab Mathematical Model

We will use the TCLab as a motivating example for this workshop.

The code below sets default font sizes.

```
import matplotlib.pyplot as plt

SMALL_SIZE = 14
MEDIUM_SIZE = 16
BIGGER_SIZE = 18

plt.rc('font', size=SMALL_SIZE)  # controls default text sizes
plt.rc('axes', titlesize=SMALL_SIZE)  # fontsize of the axes title
plt.rc('axes', labelsize=MEDIUM_SIZE)  # fontsize of the x and y labels
plt.rc('xtick', labelsize=SMALL_SIZE)  # fontsize of the tick labels
plt.rc('ytick', labelsize=SMALL_SIZE)  # fontsize of the tick labels
plt.rc('legend', fontsize=SMALL_SIZE)  # legend fontsize
plt.rc('figure', titlesize=BIGGER_SIZE)  # fontsize of the figure title
plt.rc('lines', linewidth=3)
```

Notebooks at `dowlinglab.github.io/pyomo-doe`

Modeling with Pyomo

This page is adapted from our process control class at Notre Dame; it was developed by Prof. Jeff Kantor.

```
# Install Pyomo and solvers for Google Colab
import sys

if "google.colab" in sys.modules:
    !wget "https://raw.githubusercontent.com/IDAES/idaes-pse/main/scripts/colab_helper.py"
    import colab_helper

    colab_helper.install_idaes()
    colab_helper.install_ipopt()

# Set plotting defaults
import matplotlib.pyplot as plt

SMALL_SIZE = 14
MEDIUM_SIZE = 16
BIGGER_SIZE = 18

plt.rc('font', size=SMALL_SIZE)  # controls default text sizes
```

UNIVERSITY OF NOTRE DAME

CCSI[2] Carbon Capture Simulation for Industry Impact

PROMMIS Process Optimization and Modeling for Minerals Sustainability

TC Lab: Dynamic Optimization in Pyomo

`dowlinglab.github.io/pyomo-doe/notebooks/pyomo_simulation.html`

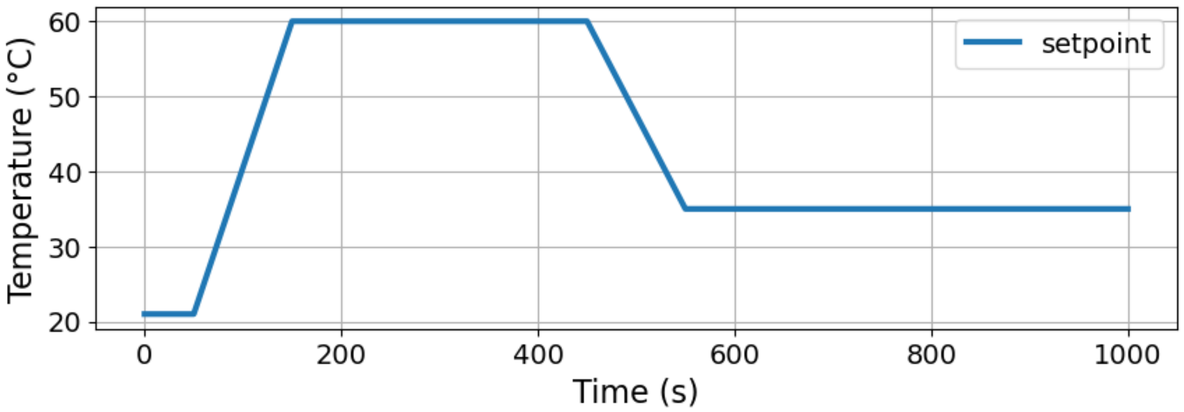
$$\min_{u(t)} \int_{t_0}^{t_f} \| SP(t) - T_H(t) \|^2 \, dt$$

$$\text{s.t.} \quad C_p^H \frac{dT_H}{dt} = U_a(T_{amb} - T_H) + U_b(T_S - T_H) + \alpha P u(t)$$

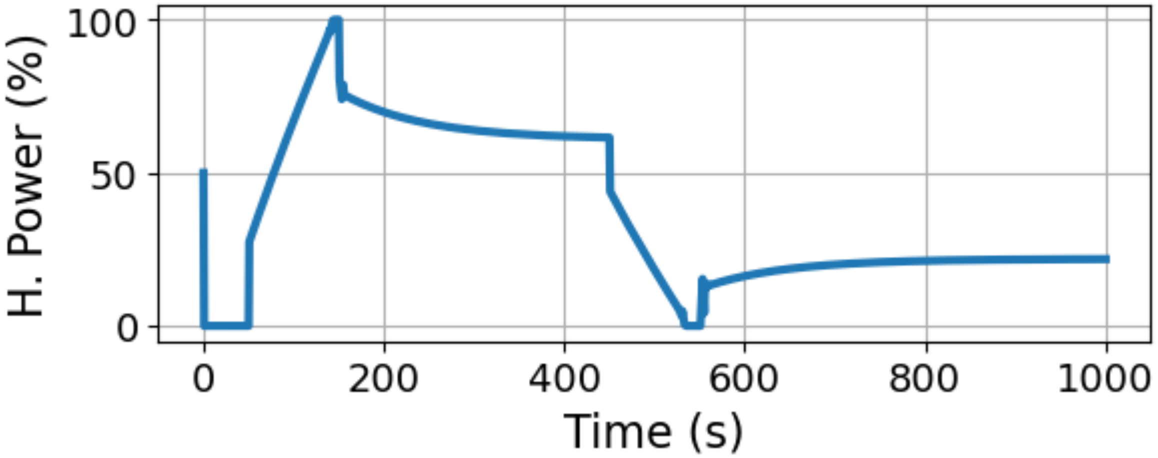
$$C_p^S \frac{dT_S}{dt} = U_b(T_H - T_S)$$

$$T_H(t_0) = T_{amb}$$

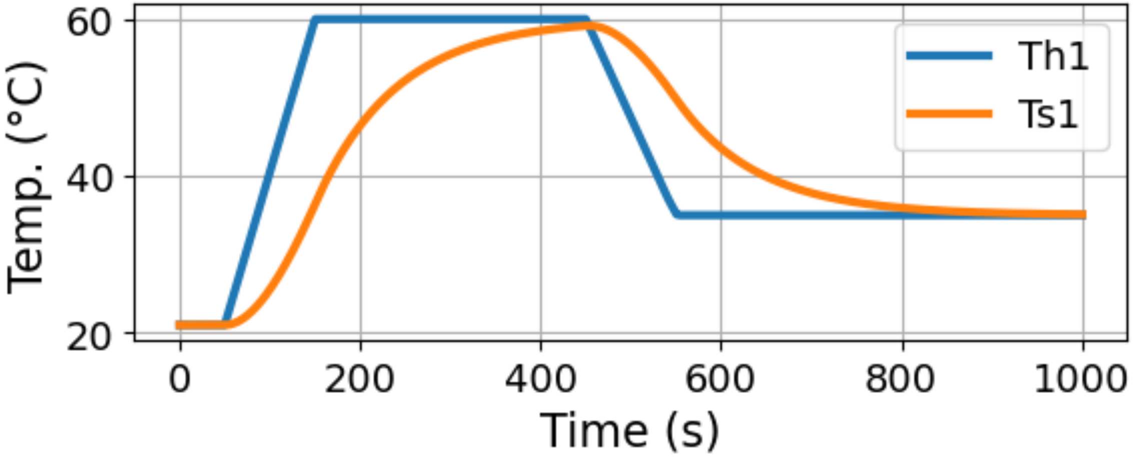
$$T_S(t_0) = T_{amb}$$



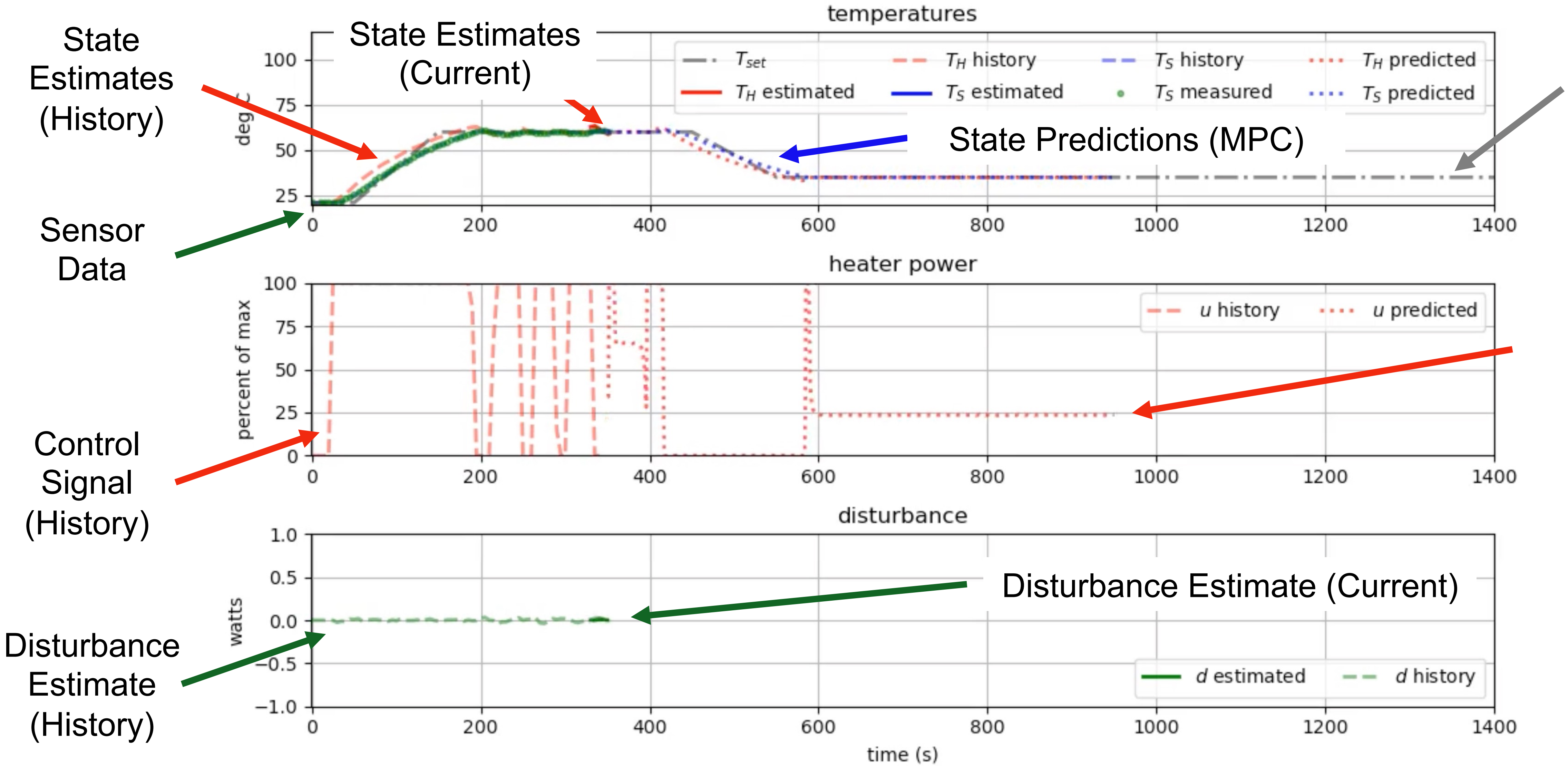
Optimal control profile



T_{h1} matches the setpoint using optimal control



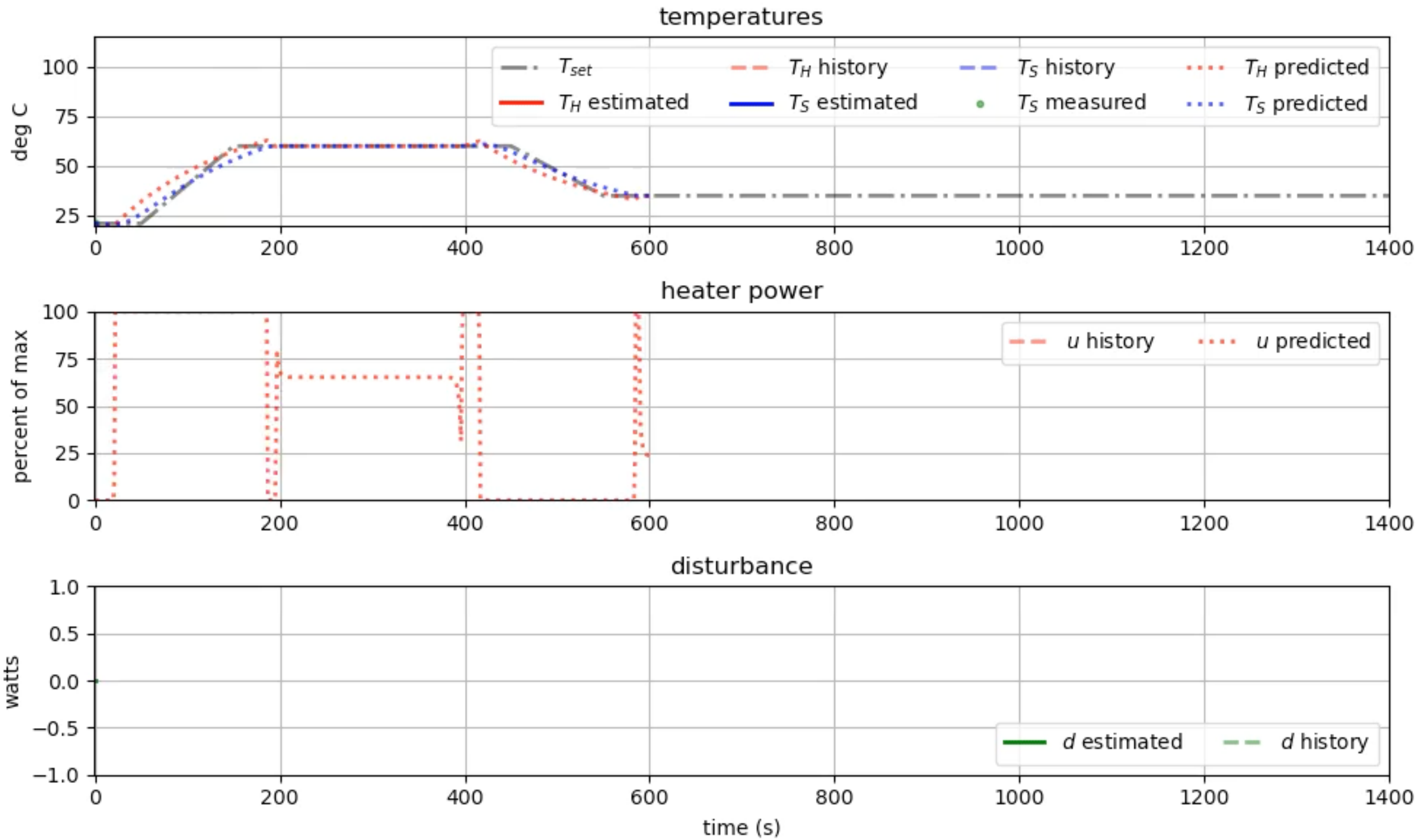
Teaching Digital Twins (MPC, State Estimation)



MPC = model predictive control

ndcbe.github.io/controls/notebooks/06-TCLab-Model-Predictive-Control.html

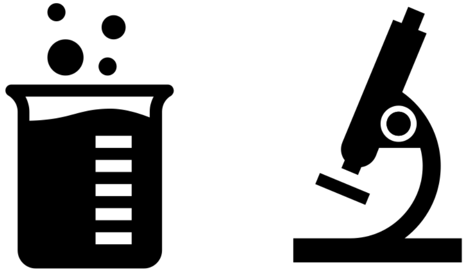
Teaching Digital Twins (MPC, State Estimation)



“Experiment” Abstraction Streamlines Closed-Loop Experiment Design

Experiment Inputs (Conditions)

Experiment (Physical)



Experiment Outputs (Data) + Measurement Uncertainty

Experiment (Digital)

ϕ : experimental design vector

$$y_i = f(x, \phi, \theta) + \varepsilon_i$$

$\hat{\boldsymbol{y}}$: measurement response variable prediction

Dr. Bethany Nicholson

Dr. John Siirola

Dr. Shawn Martin

Katherine Klise

UNIVERSITY OF NOTRE DAME

CCSI^2 Carbon Capture Simulation for Industry Impact

PROMMIS Process Optimization and Modeling for Minerals Sustainability

"Experiment" Abstraction Streamlines Closed-Loop Experiment Design

PYOMO - parmest

Known Variables

$\boldsymbol{\phi}$: experimental design vector

$\boldsymbol{y}$: measurement response variable

Experiment
(Digital)

$$y_i = f(x, \phi, \theta) + \varepsilon_i$$

Unknown Variables

$\boldsymbol{\theta}$: unknown model parameters

$$\min_{\boldsymbol{\theta}} \sum_i (y_i - \hat{y}_i)^2$$

PYOMO DoE

$\boldsymbol{\theta}$: model parameters

Experiment
(Digital)

$$y_i = f(x, \phi, \theta) + \varepsilon_i$$

$\boldsymbol{\phi}$: experimental design vector

$$\max_{\phi_l \leq \phi \leq \phi_u} \Psi\left(\mathbf{FIM}\left(\boldsymbol{\theta}, \boldsymbol{\phi}\right)\right)$$

Dr. Bethany Nicholson

Dr. John Siirola

Dr. Shawn Martin

Katherine Klise

UNIVERSITY OF NOTRE DAME

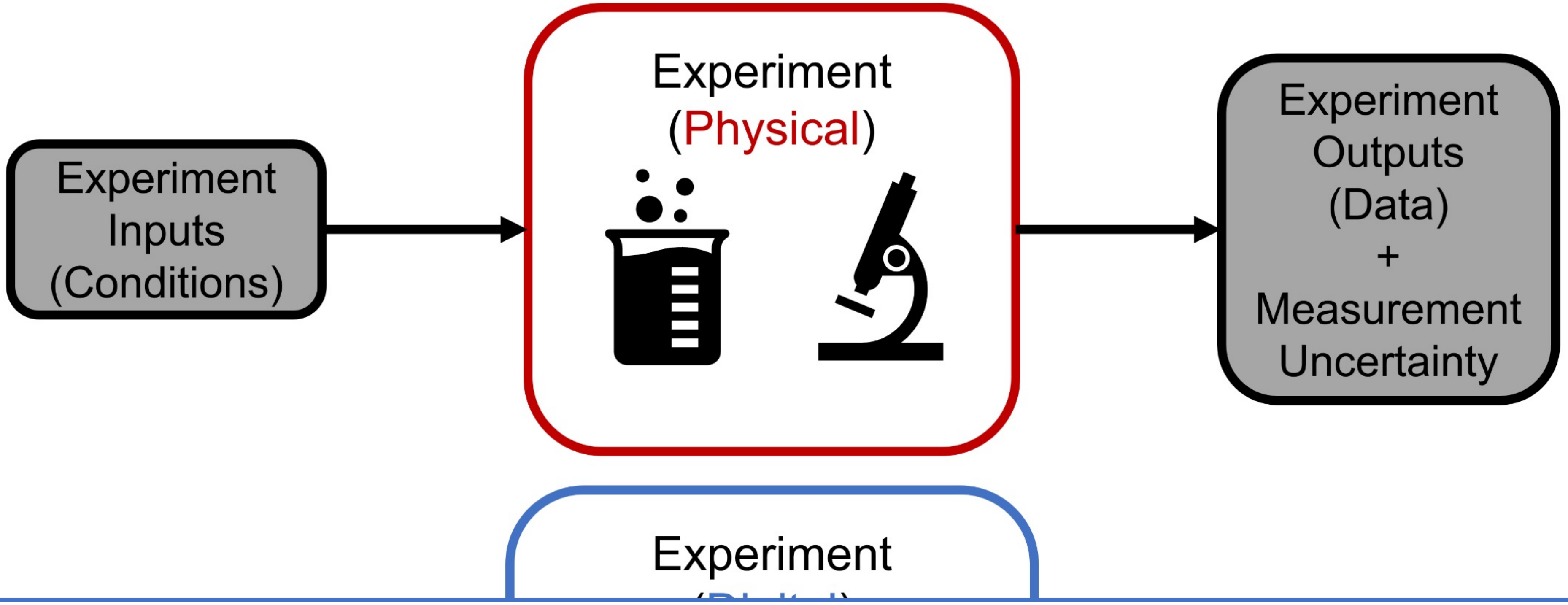
CCSI[2] Carbon Capture Simulation for Industry Impact

PROMMIS Process Optimization and Modeling for Minerals Sustainability

Notebooks at `dowlinglab.github.io/pyomo-doe`

Experiment Abstraction

The fundamental basis to parameter estimation and to optimal, science-based design of experiments (SBDoE) is an experiment. This concept is also borrows from the concept of a physical experiment:



Notebooks at `dowlinglab.github.io/pyomo-doe`

Nonlinear Regression with ParmEst

```
import sys

# If running on Google Colab, install Pyomo and Ipopt via IDAES
on_colab = "google.colab" in sys.modules
if on_colab:
    !wget "https://raw.githubusercontent.com/dowlinglab/pyomo-doe/main/notebooks/tclab_pyomo.py"

# import TCLab model, simulation, and data analysis functions
from tclab_pyomo import (
    TC_Lab_data,
    TC_Lab_experiment,
    extract_results,
    extract_plot_results,
)

# set default number of states in the TCLab model
number_tclab_states = 2
```

Parameter Estimation and Uncertainty Basics

Assume a model and error structure:

measurement

input variables

observation error (i.i.d.)

$$y_i = m(\boldsymbol{x}_i, \boldsymbol{\theta}) + \epsilon_i \qquad \epsilon_i \sim N(0, \sigma_\epsilon^2)$$

model

parameters

What values of model parameters $\boldsymbol{\theta}$ best fit the data $\boldsymbol{X}$ and $\boldsymbol{y}$?

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \Psi \coloneqq \frac{1}{2}\sum_i [y_i - m(\boldsymbol{x}_i, \boldsymbol{\theta})]^2$$

best fit estimates

How sensitive are the least-squares objective Ψ to perturbations in $\boldsymbol{\theta}$?

$$\boldsymbol{H} = \begin{bmatrix} \frac{\partial^2\Psi}{\partial\theta_1^2} & \cdots & \frac{\partial^2\Psi}{\partial\theta_n\partial\theta_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2\Psi}{\partial\theta_1\partial\theta_m} & \cdots & \frac{\partial^2\Psi}{\partial\theta_m^2} \end{bmatrix} \qquad \boldsymbol{Q}(\boldsymbol{\theta}) = \begin{bmatrix} \frac{\partial m(x_1,\boldsymbol{\theta})}{\partial\theta_1} & \cdots & \frac{\partial m(x_1,\boldsymbol{\theta})}{\partial\theta_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial m(x_n,\boldsymbol{\theta})}{\partial\theta_1} & \cdots & \frac{\partial m(x_n,\boldsymbol{\theta})}{\partial\theta_m} \end{bmatrix}$$

Hessian matrix $\boldsymbol{H} \approx \boldsymbol{Q}^T\boldsymbol{Q}$ sensitivity matrix

How does measurement uncertainty ϵ propagate into uncertainty about the regressed parameters $\hat{\theta}$?

covariance matrix for $\widehat{\boldsymbol{\theta}}$

$$\boldsymbol{V}_{\widehat{\boldsymbol{\theta}}} \approx \sigma_\epsilon^2 \boldsymbol{H}^{-1} \approx \sigma_\epsilon^2 (\boldsymbol{Q}^T\boldsymbol{Q})^{-1}$$

Fisher information matrix for $\widehat{\boldsymbol{\theta}}$

$$\boldsymbol{M}_{\widehat{\boldsymbol{\theta}}} \approx \boldsymbol{V}_{\widehat{\boldsymbol{\theta}}}^{-1} \approx \frac{1}{\sigma_\epsilon^2}(\boldsymbol{Q}^T\boldsymbol{Q})$$

Bard (1974)
Bates and Watts (1988)
Pirnay, Lopez-Negrete, Biegler (2012)

Extensions not shown: sophisticated error structures, Bayesian or MLE inference, …

TCLab: Eigendecomposition of the Fisher Information Matrix

ParmEst: `dowlinglab.github.io/pyomo-doe/notebooks/parmest.html`
FIM: `dowlinglab.github.io/pyomo-doe/notebooks/doe_exploratory_analysis.html`

```
FIM:
 [[517225.40941304   1360.01262476 -66404.72541298  -1002.47319402]
 [  1360.01262476   5004.3737258   12379.2662576    5238.40389773]
 [-66404.72541298  12379.2662576   65481.16908635  14190.01468139]
 [ -1002.47319402   5238.40389773  14190.01468139   5526.94375493]]

eigenvalues:
 [5.26802218e+05 6.26035823e+04 3.83207978e+03 1.61037063e-02]

eigenvectors:
```

U_a	[[-9.89752804e-01	-1.35949591e-01	4.36702406e-02	-7.52086327e-05]	U_a
U_b	[8.63262440e-04	-2.26164575e-01	-6.85698047e-01	-6.91857665e-01]	U_b
$1/C_p^H$	[1.42671125e-01	-9.31600001e-01	3.33329462e-01	-2.56487437e-02]	$1/C_p^H$
$1/C_p^S$	[5.79584008e-03	-2.49977462e-01	-6.45602485e-01	7.21578207e-01]]	$1/C_p^S$

Difficult to uniquely estimate U_b and C_p^S with this single experiment!

Model-Based DoE Optimization Formulation

$$\max_{\boldsymbol{\varphi}} \quad \Psi[\,\boldsymbol{M}(\,\hat{\boldsymbol{\theta}}\,,\boldsymbol{\varphi}\,)\,]$$

$$\text{s.t.} \quad \left.\begin{array}{c} \dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\,\boldsymbol{x}(t),\boldsymbol{z}(t),\boldsymbol{u}(t),\overline{\boldsymbol{w}},\hat{\boldsymbol{\theta}}\,) \\ \boldsymbol{g}(\,\boldsymbol{x}(t),\boldsymbol{z}(t),\boldsymbol{u}(t),\overline{\boldsymbol{w}},\hat{\boldsymbol{\theta}}\,) = \boldsymbol{0} \\ \boldsymbol{y}(t) = \boldsymbol{h}(\,\boldsymbol{x}(t),\boldsymbol{z}(t),\hat{\boldsymbol{\theta}}) \end{array}\right\} \text{DAE System}$$

$$\left.\begin{array}{c} \boldsymbol{f}^{\boldsymbol{0}}(\dot{\boldsymbol{x}}(t_0),\boldsymbol{x}(t_0),\boldsymbol{z}(t_0),\boldsymbol{u}(t_0),\overline{\boldsymbol{w}},\hat{\boldsymbol{\theta}}) = \boldsymbol{0} \\ \boldsymbol{g}^{\boldsymbol{0}}(\,\boldsymbol{x}(t_0),\boldsymbol{z}(t_0),\boldsymbol{u}(t_0),\overline{\boldsymbol{w}},\hat{\boldsymbol{\theta}}) = \boldsymbol{0} \\ \boldsymbol{y}^{\boldsymbol{0}}(t_0) = \boldsymbol{h}(\,\boldsymbol{x}(t_0),\boldsymbol{z}(t_0),\hat{\boldsymbol{\theta}}) \end{array}\right\} \text{Initial Conditions}$$

$$\boldsymbol{m}(\,\boldsymbol{x}(t),\boldsymbol{y}(t),\boldsymbol{z}(t),\boldsymbol{u}(t),\overline{\boldsymbol{w}},\hat{\boldsymbol{\theta}}\,) = \boldsymbol{0}$$

$\boldsymbol{y}$	Measurements (model responses)
$\hat{\boldsymbol{\theta}}$	Estimated parameters
$\boldsymbol{x}$	Time-dependent differential state variables
$\boldsymbol{z}$	Time-dependent algebraic state variables
$\boldsymbol{u}$	Time-varying control variables
$\overline{\boldsymbol{w}}$	Time-invariant control variable

Fisher information matrix (FIM):

$$\boldsymbol{M} \approx \boldsymbol{V}_{\hat{\boldsymbol{\theta}}}^{-1} \approx \sigma_\epsilon^{-2}\boldsymbol{H} \approx \sigma_\epsilon^{-2}\boldsymbol{Q}^T\boldsymbol{Q}$$

MBDoE Decisions:

$$\boldsymbol{\varphi} = (\,\boldsymbol{u}(t),\boldsymbol{x}(t_0),\boldsymbol{z}(t_0),\overline{\boldsymbol{w}},\mathbf{t}\,)$$

UNIVERSITY OF NOTRE DAME CCSI[2] Carbon Capture Simulation for Industry Impact PROMMIS Process Optimization and Modeling for Minerals Sustainability

Franceschini, G., & Macchietto, S. (2008). Model-based design of experiments for parameter precision: State of the art. *Chemical Engineering Science*, *63*(19), 4846-4872.

Alphabetic Design Criteria Measure Information Content

Figure adapted from: Franceschini, G., & Macchietto, S. (2008). *Chem. Eng. Sci.*, *63*(19), 4846-4872.

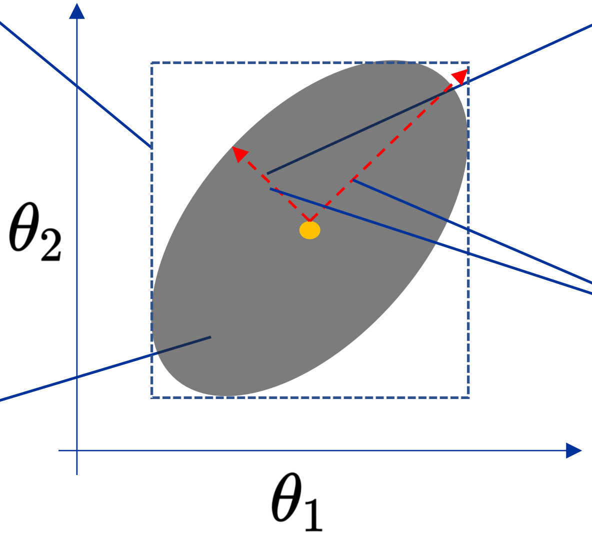
A-optimality

max trace(**M**)

enclosing box volume

poor choice for highly correlated **θ**

confidence ellipsoid for covariance matrix **V** = **M**$^{-1}$



E-optimality

max min(eig(**M**))

major axis

recommended if M is ill-conditioned

D-optimality

max det(**M**)

ellipsoid volume

robust to linear transformations

ME-optimality

min κ(**M**) = max(eig(**M**)) / min(eig(**M**))

ratio of major to minor axes

recommended if **M** is ill-conditioned

Model Discrimination

Hunter, W.G. and Reiner, A.M., 1965. Designs for discriminating between two rival models. *Technometrics*, *7*(3), pp.307-323.

Buzzi-Ferraris, G. and Forzatti, P., 1983. A new sequential experimental design procedure for discriminating among rival models. *Chemical engineering science*, *38*(2), pp.225-232.

Ferraris, G.B., Forzatti, P., Emig, G. and Hofmann, H., 1984. Sequential experimental design for model discrimination in the case of multiple responses. *Chemical engineering science*, *39*(1), pp.81-85.

Joint Parameter Precision and Model Discrimination

Alberton, A.L., Schwaab, M., Lobão, M.W.N. and Pinto, J.C., 2011. Experimental design for the joint model discrimination and precise parameter estimation through information measures. *Chemical Engineering Science*, *66*(9), pp.1940-1952.

Galvanin, F., Cao, E., Al-Rifai, N., Gavriilidis, A. and Dua, V., 2016. A joint model-based experimental design approach for the identification of kinetic models in continuous flow laboratory reactors. *Computers & Chemical Engineering*, *95*, pp.202-215.

Galvanin, F., Cao, E., Al-Rifai, N., Dua, V. and Gavriilidis, A., 2015. Optimal design of experiments for the identification of kinetic models of methanol oxidation over silver catalyst. *Chimica Oggi-Chemistry Today*, *33*(3), pp.51-56.

Pankajakshan, A., Waldron, C., Quaglio, M., Gavriilidis, A. and Galvanin, F., 2019. A Multi-Objective Optimal Experimental Design Framework for Enhancing the Efficiency of Online Model Identification Platforms. *Engineering*, *5*(6), pp.1049-1059.

UNIVERSITY OF NOTRE DAME

CCSI^2 Carbon Capture Simulation for Industry Impact

PROMMIS Process Optimization and Modeling for Minerals Sustainability

Pyomo.DoE Formulation: MBDoE as 2-Stage Program

$$\max \quad \log\det\left(\mathbf{M}(\hat{\boldsymbol{\theta}}, \boldsymbol{\varphi})\right) = 2\sum_{i=1}^{N_p} \log L_{ii}$$ D-optimality

$$\text{s.t.} \quad \mathbf{M} = \sum_{r}\sum_{r'} \tilde{\sigma}_{r,r'} \mathbf{Q}_r^{\mathrm{T}} \mathbf{Q}_{r'}$$

Stage 1

$$\mathbf{M} = \mathbf{L}\mathbf{L}^{\mathrm{T}}, \qquad L_{ii} \geq \epsilon$$ Cholesky factorization

$$q_{r,p}(t) = \frac{y_{r,p}^{+}(t) - y_{r,p}^{-}(t)}{2\,\epsilon_p}$$ Central finite difference

$$\boldsymbol{m}\left(\boldsymbol{x}_p^{+}(t), \boldsymbol{y}_p^{+}(t), \boldsymbol{z}_p^{+}(t), \boldsymbol{u}(t), \overline{\boldsymbol{w}}, \boldsymbol{\theta}_p^{+}\right) = \mathbf{0}$$

$$\boldsymbol{m}\left(\boldsymbol{x}_p^{-}(t), \boldsymbol{y}_p^{-}(t), \boldsymbol{z}_p^{-}(t), \boldsymbol{u}(t), \overline{\boldsymbol{w}}, \boldsymbol{\theta}_p^{-}\right) = \mathbf{0}$$

Two model evaluations

$$\boldsymbol{\theta}_p^{+} = \hat{\boldsymbol{\theta}} + \boldsymbol{e}_p \epsilon_p$$

$$\boldsymbol{\theta}_p^{-} = \hat{\boldsymbol{\theta}} - \boldsymbol{e}_p \epsilon_p$$

Up and down perturbations

Stage 2

$$\forall p \in \{1, \dots, N_p\}$$

Model Sensitivity

$$\mathbf{Q}_r = \begin{bmatrix} \frac{\partial y_r(t_1)}{\partial \theta_1} & \cdots & \frac{\partial y_r(t_1)}{\partial \theta_{N_p}} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_r(t_n)}{\partial \theta_1} & \cdots & \frac{\partial y_r(t_n)}{\partial \theta_{N_p}} \end{bmatrix} = [\mathbf{q}_{r,1} \quad \cdots \quad \mathbf{q}_{r,N_p}]$$

$$\mathbf{q}_{r,p} = \begin{bmatrix} \frac{\partial y_r(t_1)}{\partial \theta_p} & \cdots & \frac{\partial y_r(t_n)}{\partial \theta_p} \end{bmatrix}^{\mathrm{T}}$$

$\boldsymbol{y}$	Measurements (model responses)
$\mathbf{Q_r}$	Dynamic sensitivity for response *r*
$\boldsymbol{m}(\,)$	DAE model
$\hat{\boldsymbol{\theta}} \in \mathbb{R}^P$	Estimate for parameters
$\mathbf{M} \in \mathbb{R}^{P\times P}$	Fisher information matrix
$\mathbf{L} \in \mathbb{R}^{P\times P}$	Lower triangular Cholesky factorization
ϵ_p	Small perturbation for parameter *p*
$\boldsymbol{e}_p \in \mathbb{R}^P$	Unit vector with “1” in position *p*

Temperature Control Lab (TC-Lab) – Closed-Loop Experimental Design with New Experiment Abstraction

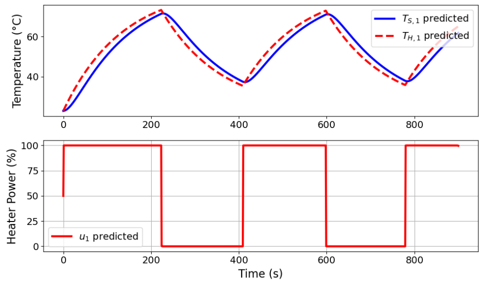
Parameter Estimation

$$C_p^H \frac{dT_{H,1}}{dt} = U_a(T_{amb} - T_{H,1}) + U_b(T_{S,1} - T_{H,1}) + \alpha P_1 u_1$$
$$C_p^S \frac{dT_{S,1}}{dt} = U_b(T_{H,1} - T_{S,1})$$

$$\theta \equiv \{C_P^H, C_P^S, U_a, U_b\}$$
$$y \equiv \{T_{S,1}\}$$
$$\phi \equiv \{u_1\}$$

PYOMO - parmest

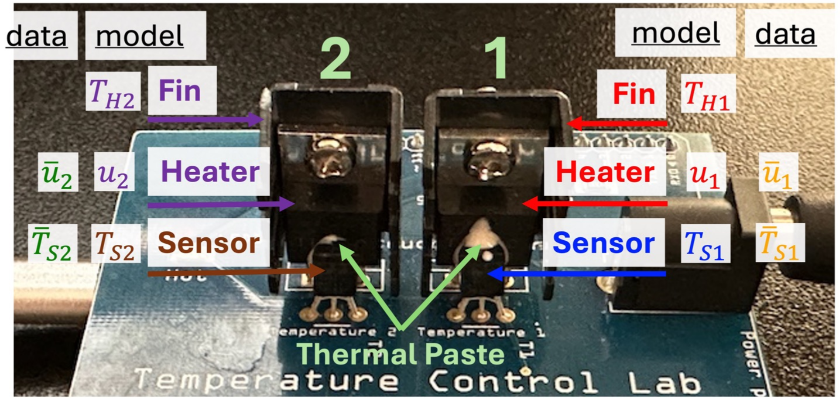
Optimal Experiment Design/Optimal Control



Digital Components

Physical Components

Physical System



UNIVERSITY OF NOTRE DAME

CCSI2 Carbon Capture Simulation for Industry Impact

PROMMIS Process Optimization and Modeling for Minerals Sustainability

Notebooks at `dowlinglab.github.io/pyomo-doe`

Pyomo.DoE: Exploratory Analysis

```
import matplotlib.pyplot as plt

SMALL_SIZE = 14
MEDIUM_SIZE = 16
BIGGER_SIZE = 18

plt.rc('font', size=SMALL_SIZE)  # controls default text sizes
plt.rc('axes', titlesize=SMALL_SIZE)  # fontsize of the axes title
plt.rc('axes', labelsize=MEDIUM_SIZE)  # fontsize of the x and y labels
plt.rc('xtick', labelsize=SMALL_SIZE)  # fontsize of the tick labels
plt.rc('ytick', labelsize=SMALL_SIZE)  # fontsize of the tick labels
plt.rc('legend', fontsize=SMALL_SIZE)  # legend fontsize
plt.rc('figure', titlesize=BIGGER_SIZE)  # fontsize of the figure title
plt.rc('lines', linewidth=3)
```

```
import sys

# If running on Google Colab, install Pyomo and Ipopt via IDAES
on_colab = "google.colab" in sys.modules
```

TC Lab: DoE, Exploratory Analysis

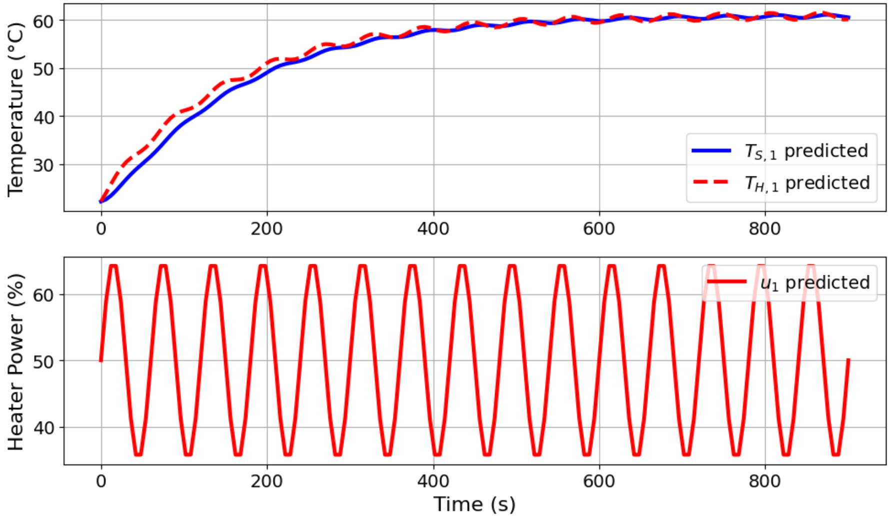
dowlinglab.github.io/pyomo-doe/notebooks/doe_exploratory_analysis.html

Sensitivity of the FIM to experimental design.

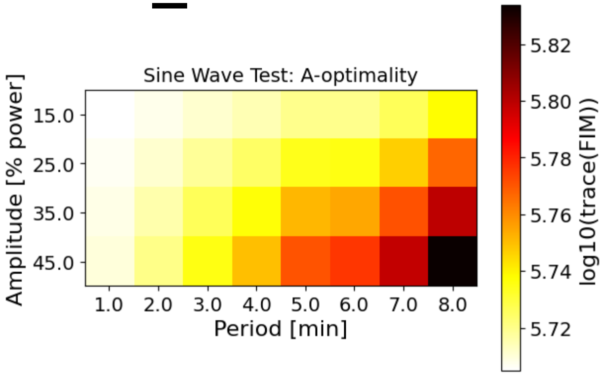
Example: Sine Wave

Vary:

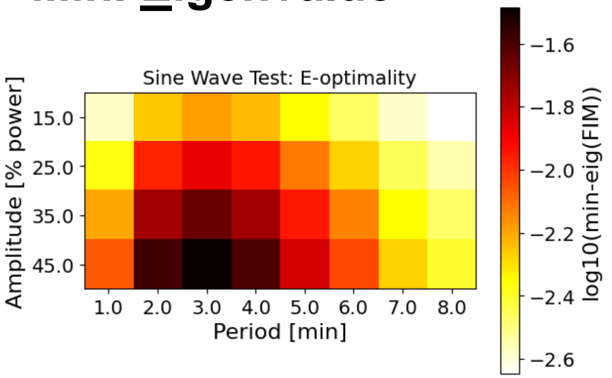
- period (from 1 to 8 minutes)
- amplitude (from 15% to 50%)



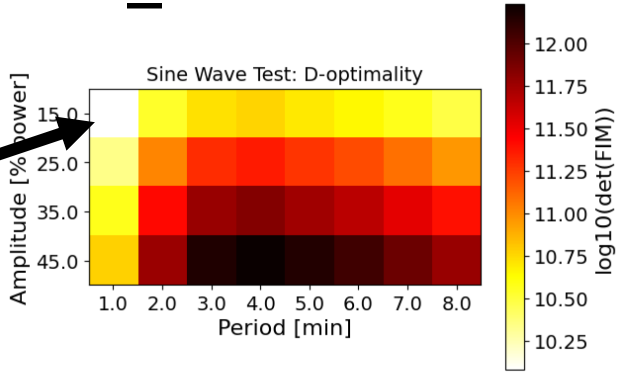
trAce



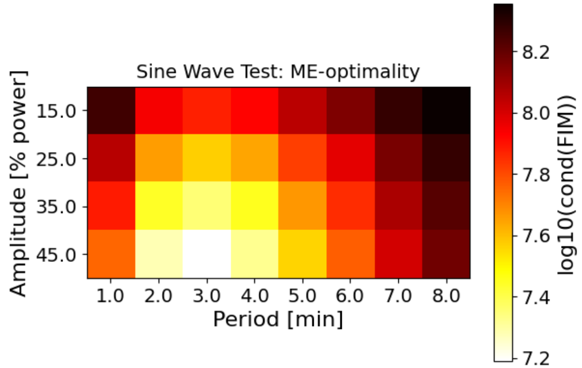
Min. Eigenvalue



Determinant



Condition Number



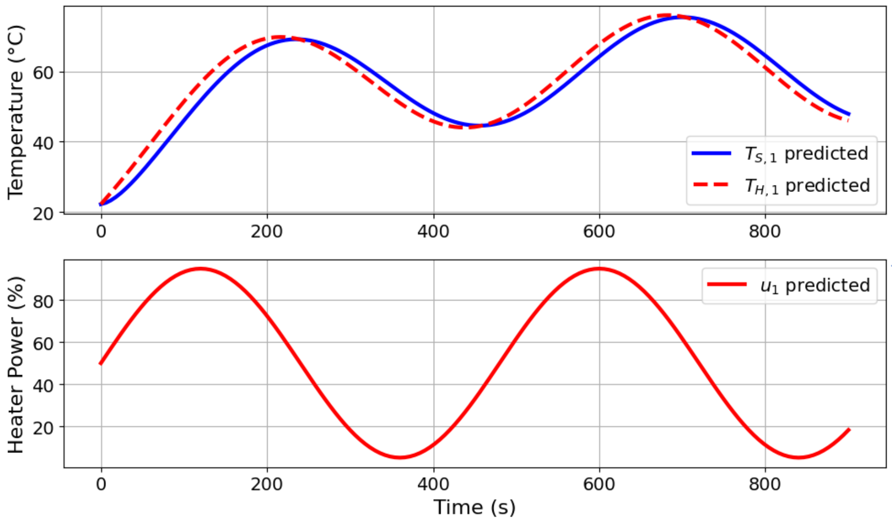
UNIVERSITY OF NOTRE DAME

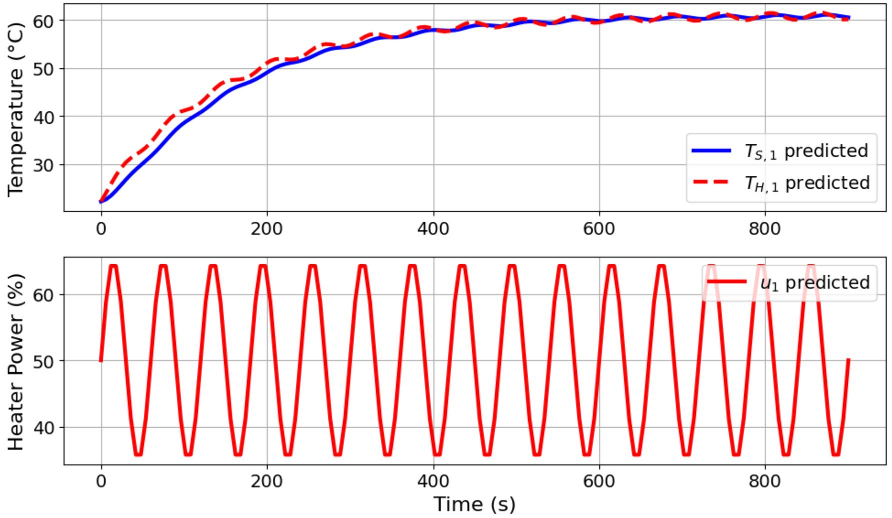
CCSI[2] Carbon Capture Simulation for Industry Impact

PROMMIS Process Optimization and Modeling for Minerals Sustainability

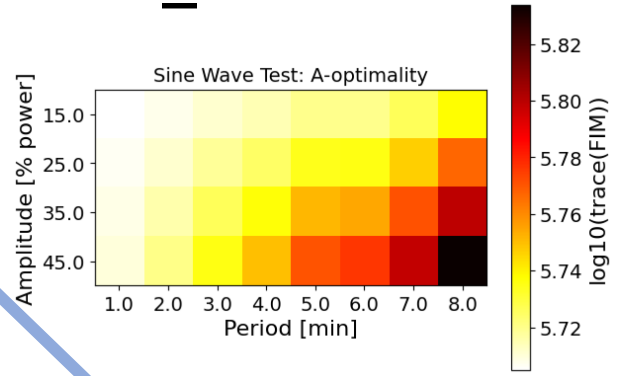
TC Lab: DoE, Exploratory Analysis

dowlinglab.github.io/pyomo-doe/notebooks/doe_exploratory_analysis.html

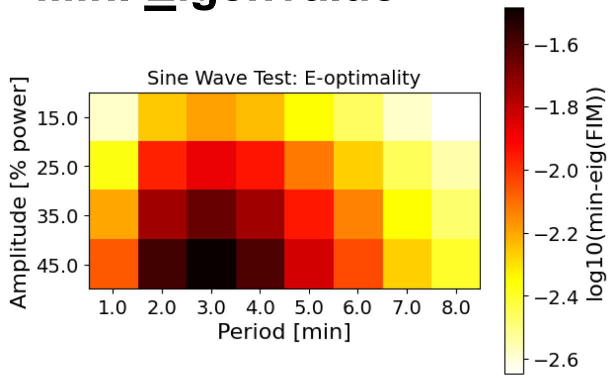




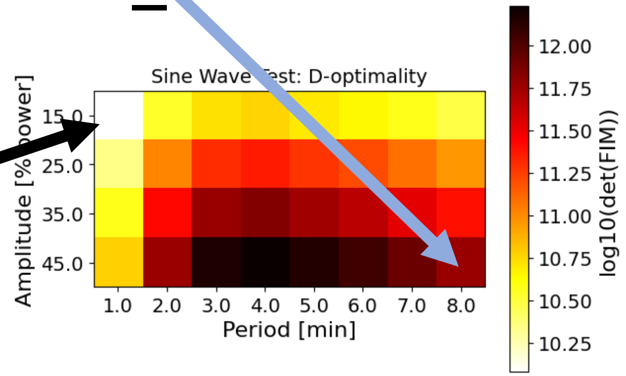
trAce



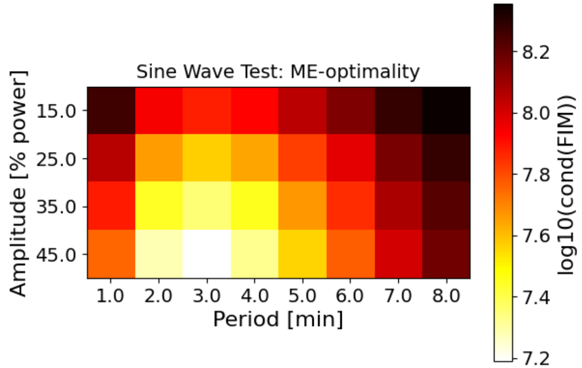
Min. Eigenvalue



Determinant



Condition Number



UNIVERSITY OF NOTRE DAME

CCSI[2] Carbon Capture Simulation for Industry Impact

PROMMIS Process Optimization and Modeling for Minerals Sustainability

Notebooks at `dowlinglab.github.io/pyomo-doe`

Pyomo.DoE: Optimization

Our earlier exploratory analysis showed the sine wave experiment alone is rank deficient. What if instead of optimizing the sine wave parameters a and p, we directly optimize $u(t)$. In other words, we will formulate model-based design of experiments as an optimal control problem.

Maximize a scalar-valued function $\psi(\cdot)$ of the Fisher information matrix $\mathbf{M}$:

$$
\begin{aligned}
\max_{u} \quad & \psi(\mathbf{M}(u) + \mathbf{M}_0) \\
\text{s.t.} \quad & C_p^H \frac{dT_H}{dt} = U_a(T_{amb} - T_H) + U_b(T_S - T_H) + \alpha P u(t) \\
& C_p^S \frac{dT_S}{dt} = U_b(T_H - T_S) \\
& 0\% \leq u(t) \leq 100\% \\
& T_H(t_0) = T_{amb} \\
& T_S(t_0) = T_{amb}
\end{aligned}
$$

TC Lab: A-Optimal Next Experiment

dowlinglab.github.io/pyomo-doe/notebooks/doe_optimize.html

$$\max_{u} \ \log \text{trace}\left(\mathbf{M}(u) + \mathbf{M}_0\right)$$

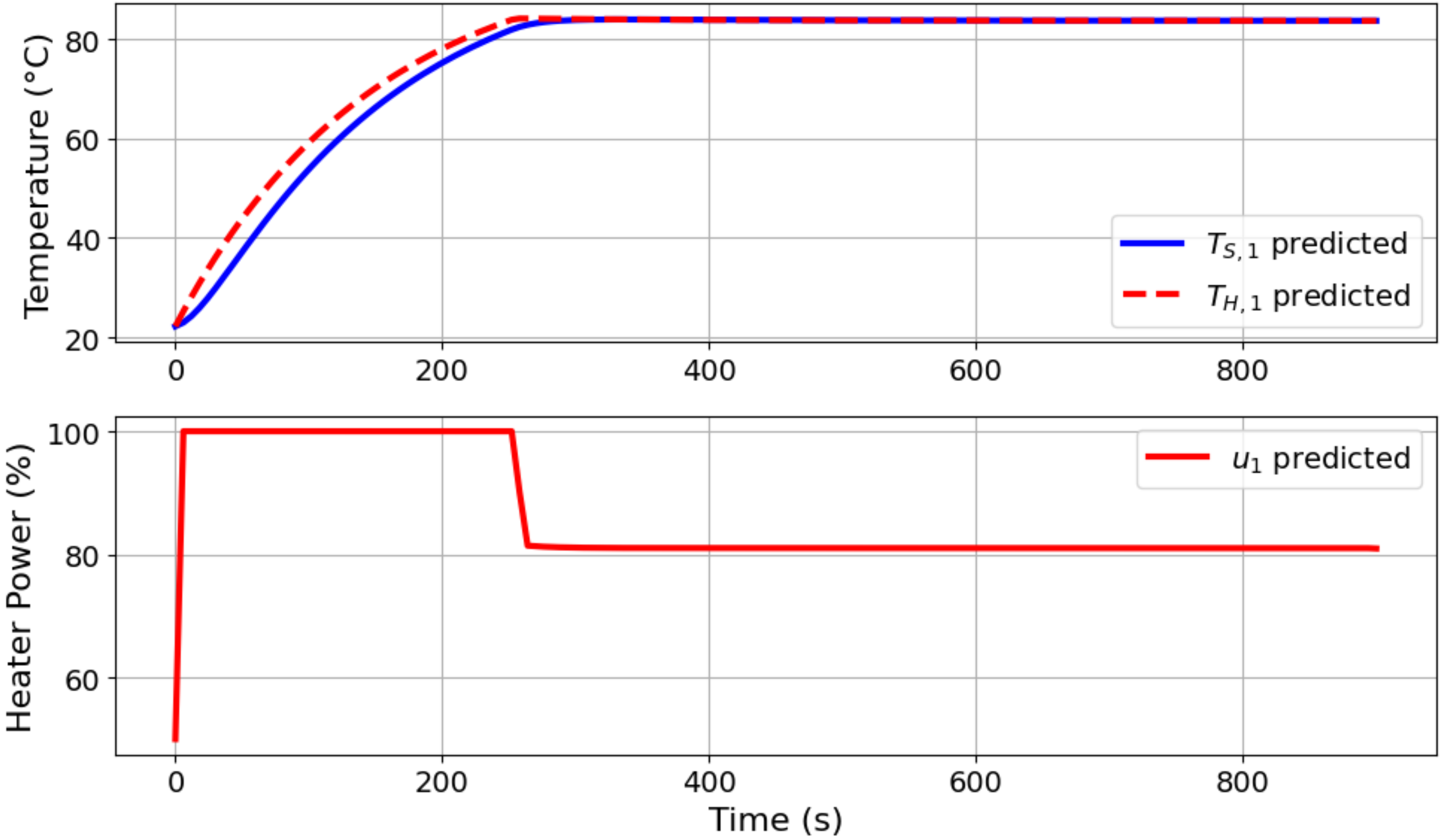
$$\text{s.t.} \quad C_p^H \frac{dT_H}{dt} = \ldots$$

$$C_p^S \frac{dT_S}{dt} = \ldots$$

$$0\% \leq u(t) \leq 100\%$$

$$T_H(t_0) = T_{amb}$$

$$T_S(t_0) = T_{amb}$$



UNIVERSITY OF NOTRE DAME
CCSI²
Carbon Capture Simulation for Industry Impact
PRO
Process Optimizatio
for Minerals Sustain

$U_b C_p^S$ is more estimable with two experiments (sine wave test, A-optimal)!

TC Lab: D-Optimal Next Experiment

dowlinglab.github.io/pyomo-doe/notebooks/doe_optimize.html

Best Sine Wave

$\log_{10}(\det(M)) = 12$

Optimal Control

$\log_{10}(\det(M)) = 14.6$

$$\max_{u} \quad \log\det(\mathbf{M}(u) + \mathbf{M}_0)$$

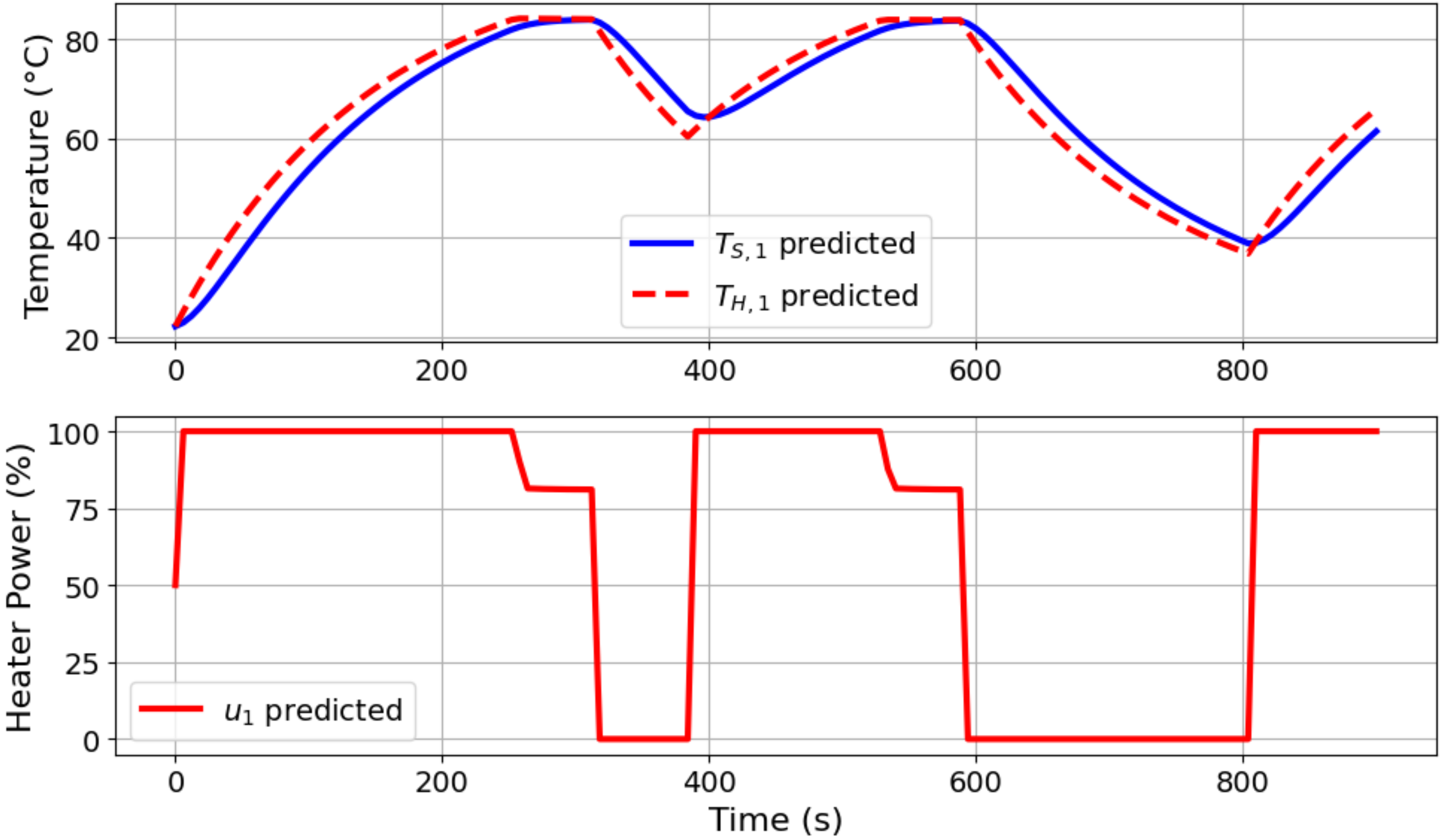
$$\text{s.t.} \quad C_p^H \frac{dT_H}{dt} = \ldots$$

$$C_p^S \frac{dT_S}{dt} = \ldots$$

$$0\% \leq u(t) \leq 100\%$$

$$T_H(t_0) = T_{amb}$$

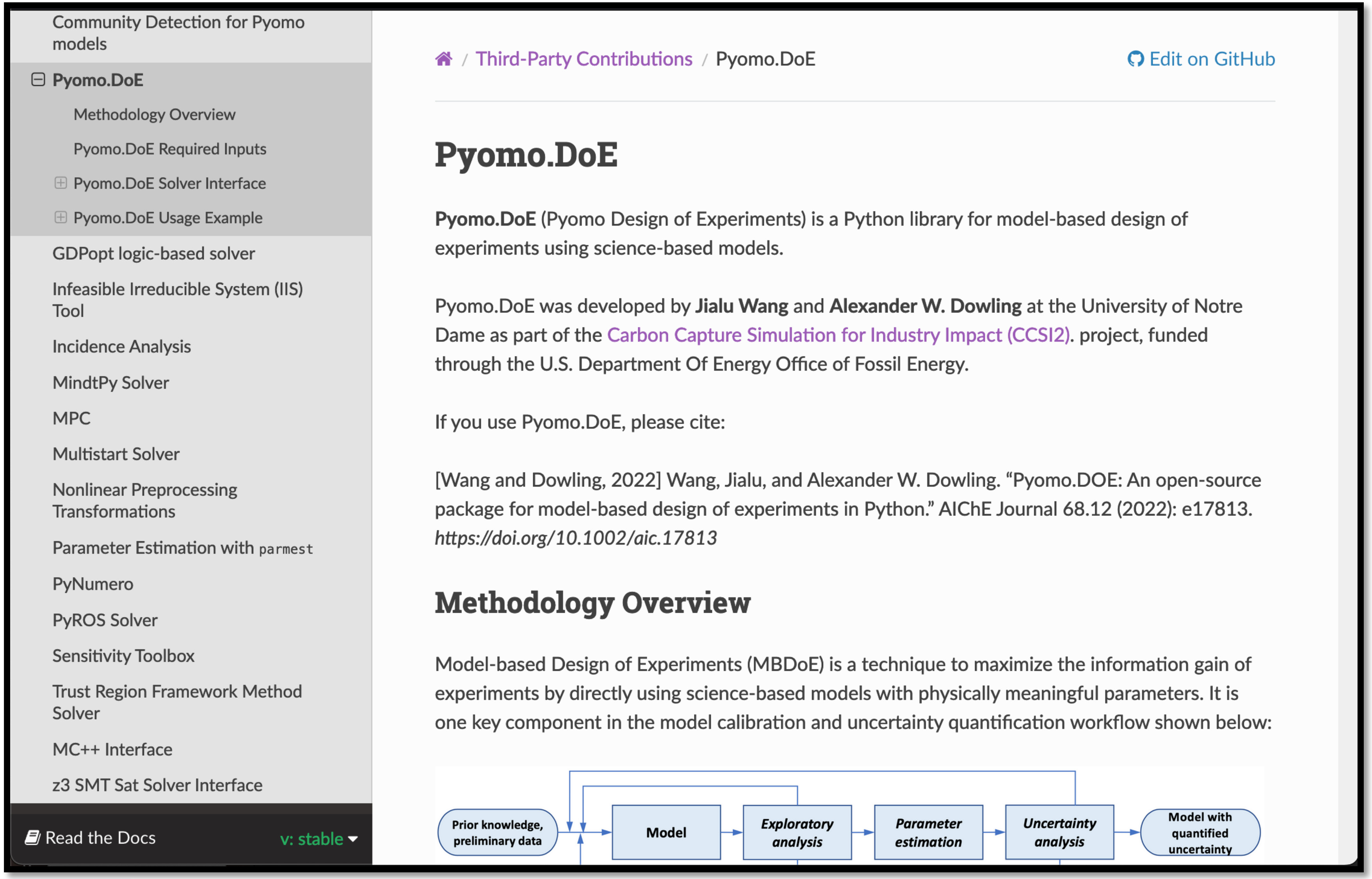
$$T_S(t_0) = T_{amb}$$



$U_b C_p^S$ is more estimable with two experiments (sine wave test, D-optimal)!

Getting Started with Pyomo.DoE

Documentation: https://pyomo.readthedocs.io/en/stable/contributed_packages/doe/doe.html



Community Detection for Pyomo models
Pyomo.DoE
Methodology Overview
Pyomo.DoE Required Inputs
Pyomo.DoE Solver Interface
Pyomo.DoE Usage Example
GDPopt logic-based solver
Infeasible Irreducible System (IIS) Tool
Incidence Analysis
MindtPy Solver
MPC
Multistart Solver
Nonlinear Preprocessing Transformations
Parameter Estimation with `parmest`
PyNumero
PyROS Solver
Sensitivity Toolbox
Trust Region Framework Method Solver
MC++ Interface
z3 SMT Sat Solver Interface

Read the Docs v: stable

/ Third-Party Contributions / Pyomo.DoE — Edit on GitHub

Pyomo.DoE

Pyomo.DoE (Pyomo Design of Experiments) is a Python library for model-based design of experiments using science-based models.

Pyomo.DoE was developed by **Jialu Wang** and **Alexander W. Dowling** at the University of Notre Dame as part of the Carbon Capture Simulation for Industry Impact (CCSI2). project, funded through the U.S. Department Of Energy Office of Fossil Energy.

If you use Pyomo.DoE, please cite:

[Wang and Dowling, 2022] Wang, Jialu, and Alexander W. Dowling. "Pyomo.DOE: An open-source package for model-based design of experiments in Python." AIChE Journal 68.12 (2022): e17813. *https://doi.org/10.1002/aic.17813*

Methodology Overview

Model-based Design of Experiments (MBDoE) is a technique to maximize the information gain of experiments by directly using science-based models with physically meaningful parameters. It is one key component in the model calibration and uncertainty quantification workflow shown below:

ParmEst and Pyomo.DoE Development Plans

Coming soon:

- Improved initialization
- Improved optimization performance
  - NLP decomposition
  - Grey-box objective calculations
- Improved modeling abstraction
  - Multiple experiments (e.g., planning batches)
  - Parameter uncertainty
- More applications, examples, and collaborations
- End-to-end uncertainty workflow (via interface with PyROS)

Acknowledgements

Pyomo.DoE development: *We graciously acknowledge funding from the U.S. Department of Energy, Office of Fossil Energy and Carbon Management, through the Carbon Capture Program (CCSI²) and Office of Resource Sustainability (PrOMMiS).*

This project was funded by the Department of Energy, National Energy Technology Laboratory an agency of the United States Government, through a support contract. Neither the United States Government nor any agency thereof, nor any of their employees, nor the support contractor, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

PYOMO DoE

PROMMIS Process Optimization and Modeling for Minerals Sustainability

CCSI² Carbon Capture Simulation for Industry Impact

Pyomo Team (SNL):
- Bethany Nicholson
- John Siirola
- Miranda Mundt

Contributors (ND):
- Dr. Jialu Wang
- Dr. Dan Laky
- Hailey Lynch

Thank you to Prof. Jeff Kantor (1954-2023) for the TCLab examples and so much more.